1. Introduction

1.1 Purpose

The purpose of this document is to provide a comprehensive description of the myTaxiService system. It’s aim is to describe what the software should do and identify the capabilities and characteristics of the system being developed. This document is meant for everyone included in the production of the software.

1.2 Scope

The project is about developing an application that will enable fast and optimized taxi services in the city. The application will allow users to register and then sign in into the app for using its services. Also taxi drivers can register and sign into the myTaxiService application with the purpose to manage their availability and duties.

Once registered into the app, the **user** can do the following actions:

* request a taxi
* make reservation
* cancel reservation
* find available taxi near
* receive confirmation about the vehicle, its code and estimated waiting time
* manage user profile
* report driver

Also the registered **taxi driver** once signed in can do these actions:

* inform the system about availability
* confirm/decline a request for taxi call
* manage driver profile
* report user

1.3 Glossary

MakeReservation Passenger request a vehicle at least 2 hours before the ride

Request Passenger request an immediate ride

RideConfirmation Receive a confirmation about the confirmed ride with all the information about the particular ride

ReservationConfirmation Notification that the reservation is successfully completed

ReportDriver Passenger reports driver in case of any irregularities during the ride

ReportUser Taxi driver reports user (particular passenger) in case of any irregularities during the ride

Guest Not registered person that visits the app

User A person that is already registered and signed in as user

TaxiDriver A person that is already registered and signed in as driver

API Application Programming Interface

GPS Global Positioning System

2. Overall Description

2.1 Product perspective

Our mobile web application will provide the user with services described in section 1.2. The software will be developed using a client-server model. The server side contains the application logic and is used to interact with permanent storage, serve pages to the client and process user input. The web client consists of dynamic web pages which provides user friendly graphical interface and the web browser through which they are accessed. The application is platform independent. The only requirement is users having a web browser installed on a device of their choice.

2.2 Identifying stakeholders

There are four distinct interest groups of people regarding this project:

* *Company -* which required project specification and expects it to be delivered in a way that satisfies given specification while respecting the set deadlines and budget
* *Developer group -* in this case group of two people
* *Taxi driver -* worker at the company that ordered the software product
* *Passenger* - person who need a ride to specific location

2.3 User characteristics

myTaxiService is expected to have users across a wide range of demographics, meaning users of any age, gender and educational background. Still it is assumed that people using our software do have the basic web browsing skills.

2.4 Actors identifying

There are four possible actors interacting with our system:

* *Guest* - is a person that access the system but has never registered on it or still hasn`t logged in. The only possibilities of the guest, who can access just the initial page, are to register into it or to log in.
* *User* - is a person that is already registered and logged into the system. The user can access and manage with all the services that the application offers.
* *Taxidriver -* a person that is already registered and logged into the system. The user can access and manage with all the services that the taxidriver application offers.
* *Admin* - is the person that is responsible for handling reports on users and drivers of my taxi service. Admin can ban a user or a driver from the system.

2.5 Goals

Having possible users in mind, myTaxyService should have these features:

* registering a new user
* sending notification about taxi availability
* confirming about the reserved vehicle, its code and waiting time
* managing user profile
* ban user/driver
* for each **user** it should provide:
* logging in to his/her profile
* making reservation
* requesting taxi
* canceling reservation
* reporting driver
* for each **taxi driver** it should provide:
* logging in to his/her profile
* confirming/declining a request for taxi call
* reporting user

2.6 Domain properties

* user making reservation from a specific location to a specific destination
* taxis are organized into taxi zones
* the payment process is irrelevant to the system, it`s done between the passenger and driver

2.7 Assumptions

Considering that there were some ambiguities in the specification document, the following facts are assumed:

* user registers with email and password
* user can change the email and password
* user can only have one account
* there is a Terms & Conditions section that indicates clearly the usage of the application, which if not followed will result with deactivation of the account
* we assume that Google Maps service will calculate location used by myTaxiService accurately
* if Google Maps does not provide automatically the location, new screen is showed to the user where he/she types the address
* if the taxi driver does not respect the estimated waiting time, he is banned from the system
* is there are any irregularities with the taxi driver, new vehicle is sent to the passenger

3. Requirements

3.1 Functional requirements

Functional requirements are defined for the system and for each actor defined in section 2.4.

For a Guest the system should provide two functionalities:

* Register: the guest is on the initial page of the application and should provide some information (name, surname, email, password) in order to register to the app
* Log in: after entering the correct identification information the guest will access the homepage of the app

For a **User** the system should provide the following functionalities:

* Option to sign out
* Manage profile setting:
* change email
* change password
* Request taxi
* Make reservation
* Cancel reservation
* Report driver

For a **TaxiDriver** the system should provide the following functionalities:

* Option to sign out
* Manage profile setting:
* change email
* change password
* Confirming/declining a taxi call
* Managing the requests
* Report user

The system should:

* send notifications about taxi availability
* confirm about the reserved vehicle, its code and waiting time
* ban user/driver

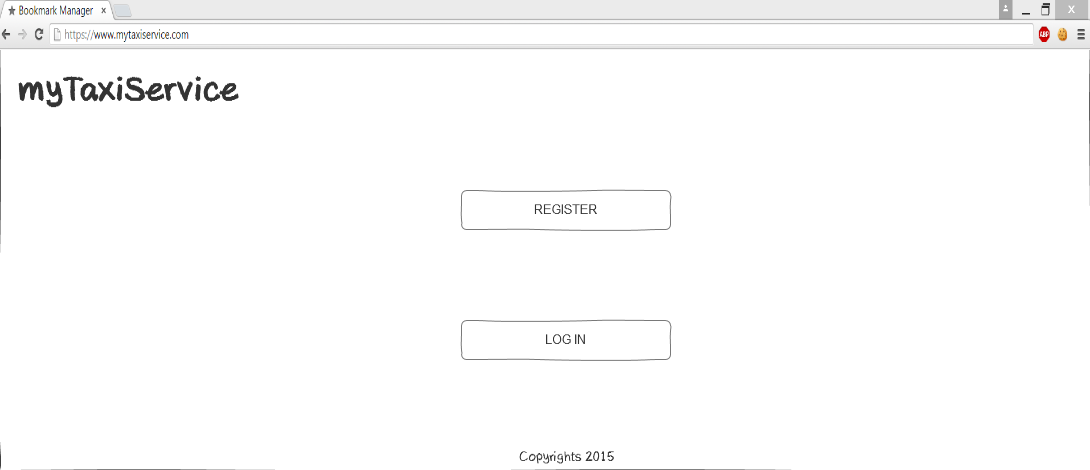
3.2 Non-functional requirements

**3.2.1 User-interface** of the application should be intuitive, simple and easy to use even for non-technically savvy users. The design should be clean and user friendly.

**Availability -** the application should be easily accessible at any time by the user who is using any kind of device with web browser installed on it.

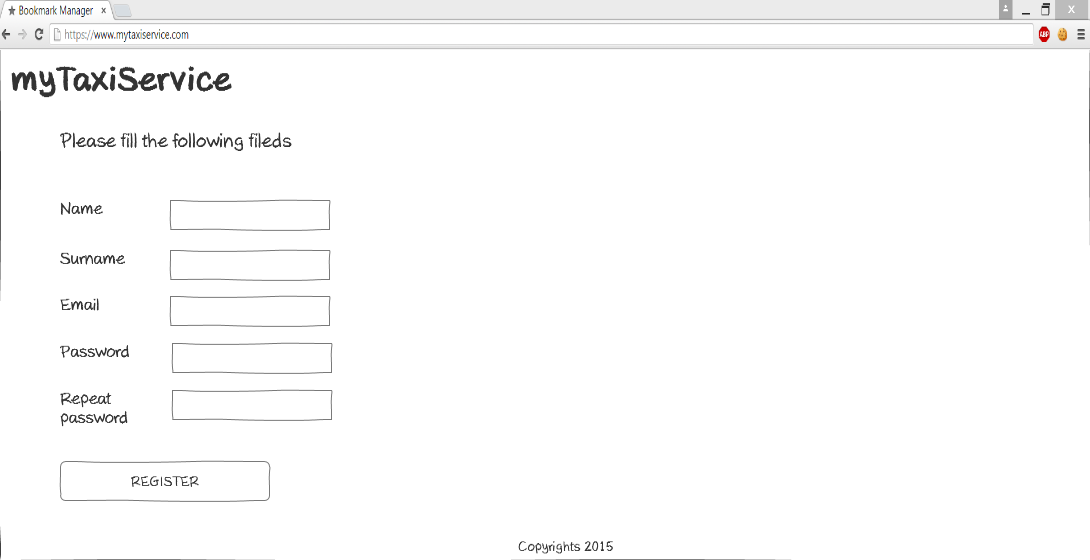
In order how we imagine the interface of our application we introduce a brief hand sketching of the main pages that the app will have.

* The following hand sketching represent the actions that the guest and user can do.



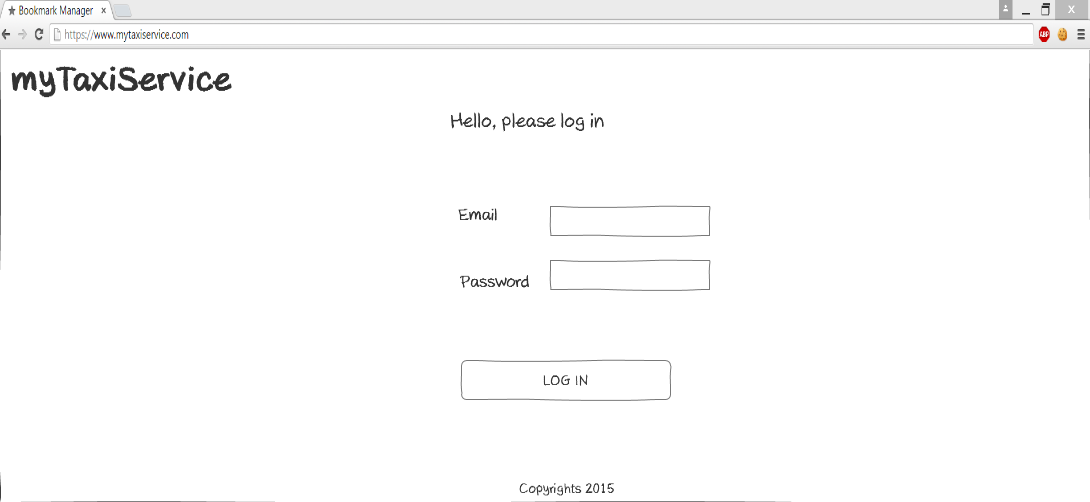
Img 1. Initial page of the app

The initial page is accessible by the guest, any person that is not registered or logged in into the app. There are only two options in this page either to register to the app or to log in into an existing profile.



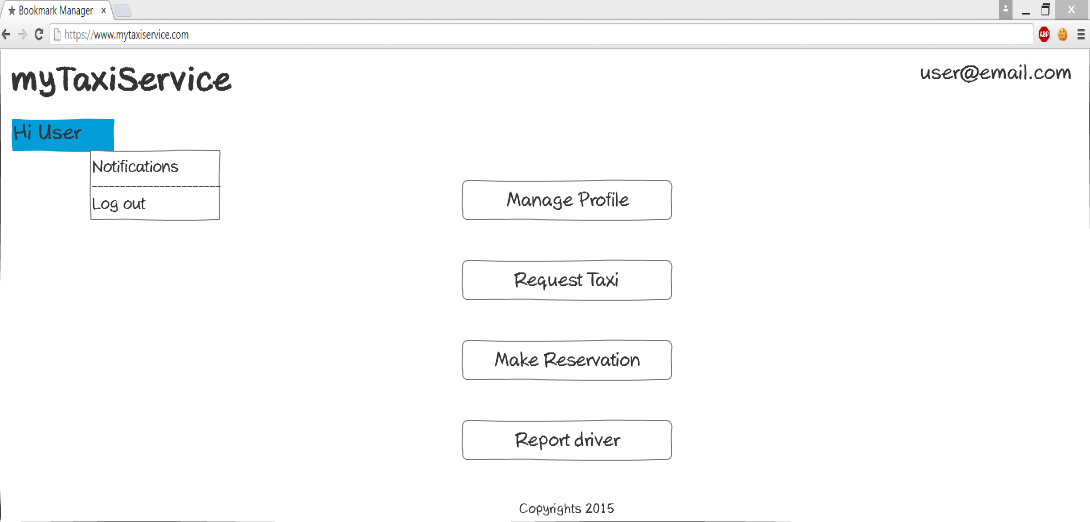
Img 2. Register page

If the person, who is initially a guest, chooses the register button it redirects him/her to the registration page. This is the page where the user is obligated to type his name, surname, email, and password into the given fields. Then he clicks on the register button and if the data is correct, he/she is redirected to the homepage of the app.



Img3. Login page

In the log in page the user should provide his/her email and password in order to enter to his/her profile and access the services that the app provides.

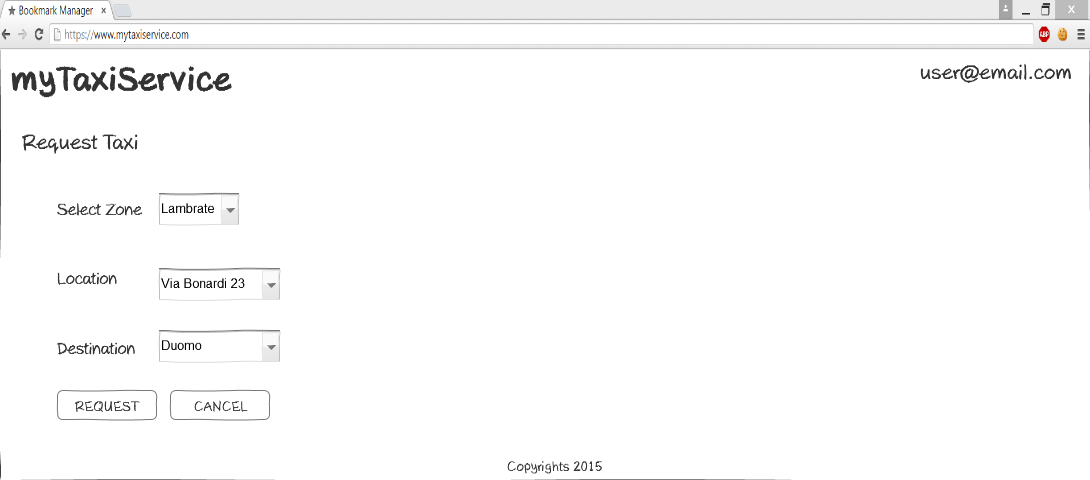


Img 4. Homepage

Once logged in, the user is lead to the homepage of the app where he can access the services that this app provides. He can choose either to manage the profile, request a taxi, or make a reservation.

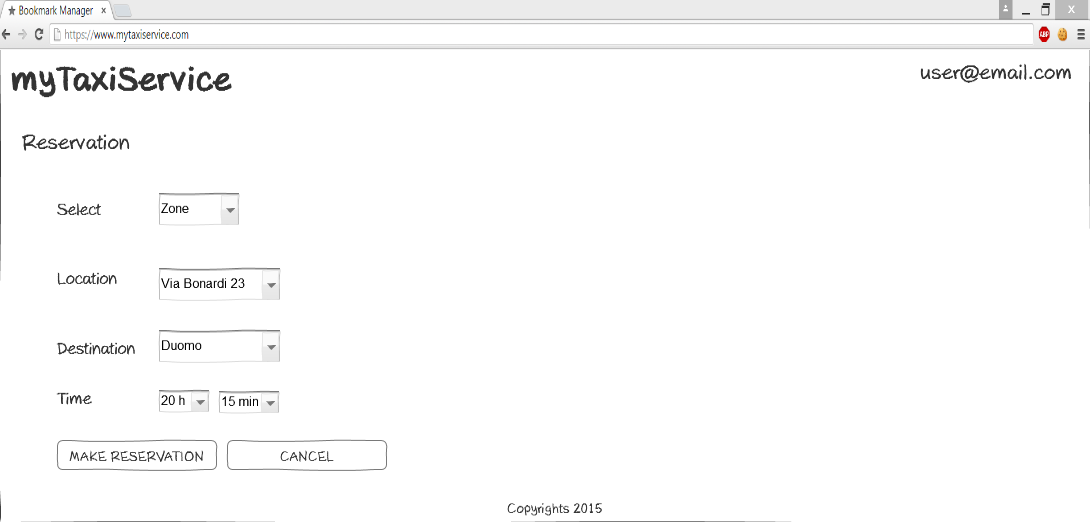


Img 5. Manage profile page

When the user clicks on the manage profile button, the app will bring him/her to the manage profile page where he/she can modify the email and password that were previously used. 

Img 6. Request taxi page

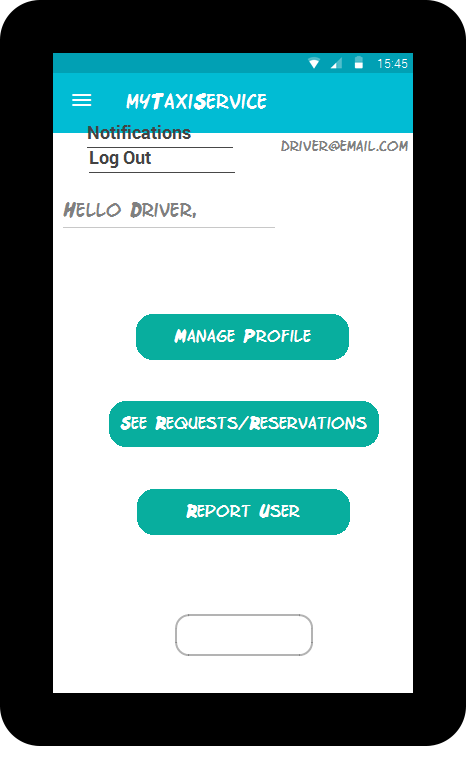
If the user wants to request a taxi immediately, he will click on the Request taxi button. In this page he should specify the following information: zone, current location, and destination of the ride. Then he clicks request and the request is sent to the system.



Img 7. Make reservation page

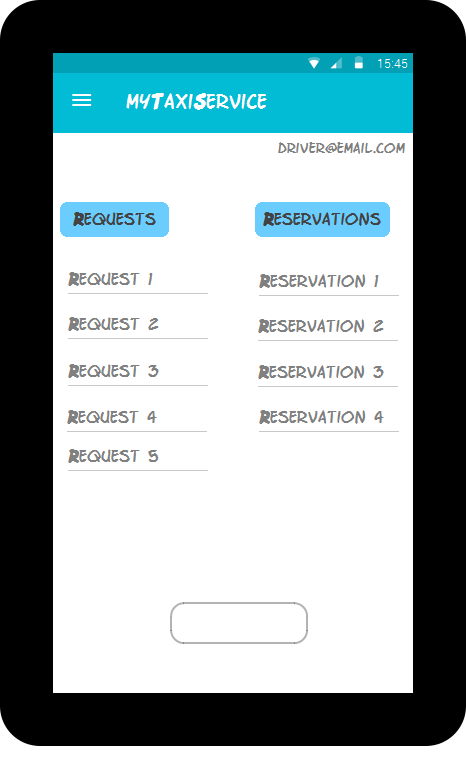
If the user wants to make a reservation for a tax, he will click on the Make reservation button. In this page he should specify the following information: zone, current location, destination of the ride, and the time when he wants use the service, which should be at least 2 hours before the ride. Then he clicks make reservation and the reservation is sent to the system.

* Next hand sketchings represent the actions that the driver can do.



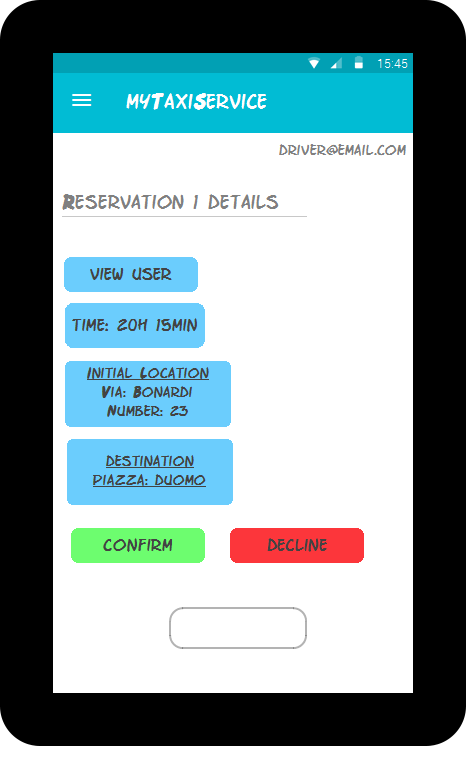
Img 8. Driver homepage

Once logged in, the driver is lead to the homepage of the app where he can access the services that this app provides. He can choose either to manage the profile, see the requests or the reservations on which he should respond.



Img 9. Notification page

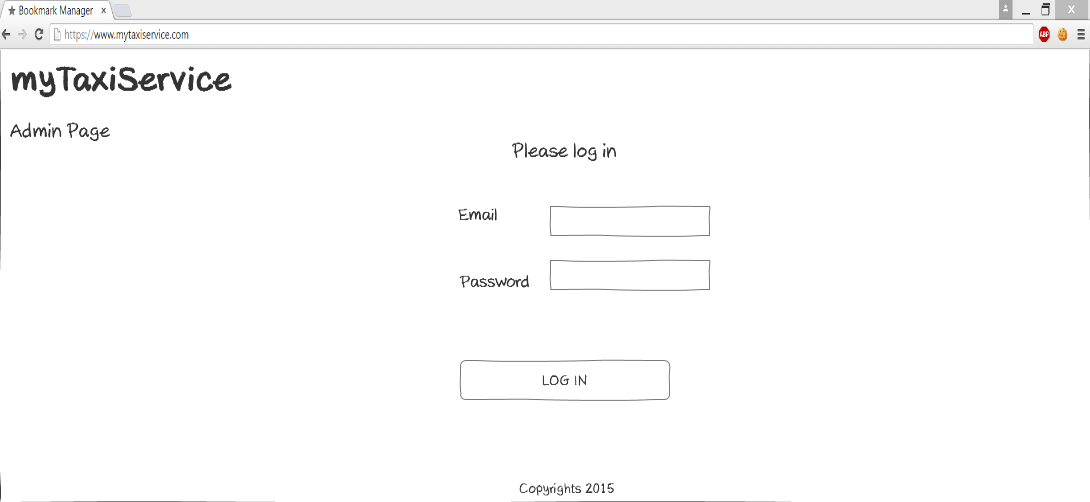
When the driver chooses See requests/reservations, the page where all the requests and reservations are, is loaded. If he wants to see the details and to respond to a particular request/reservation he should click on it.



Img 10. Reservation details page

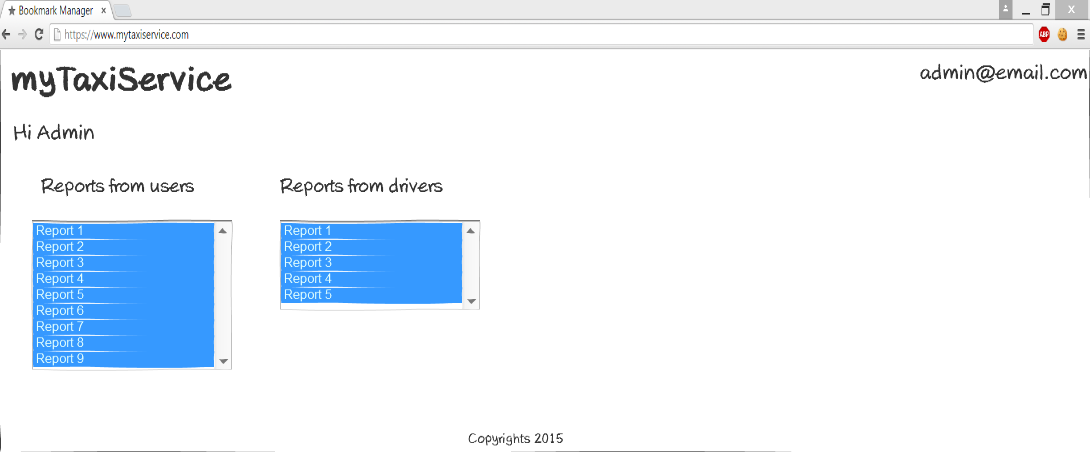
When driver chooses some reservation from the list, its details are represented in a new page. Here he can see who is making the reservation (link to the user profile is provided), the time that the vehicle should be on the starting location, the address of the location (starting point), and the final destination. Further he can confirm or decline the reservation.

* And last are the hand sketching for the actions that the admin can do.



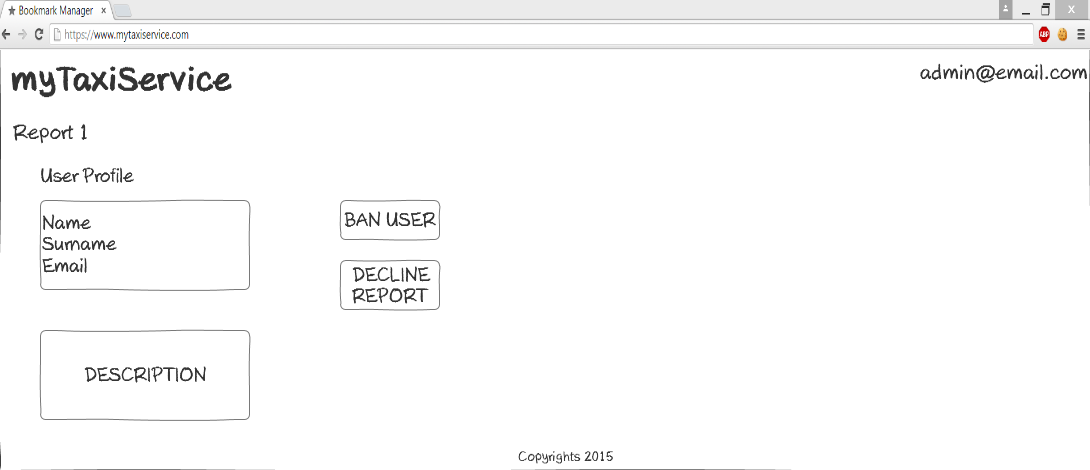
Img 11. Admin log in page

In the admin log in page the admin should provide his/her email and password in order to enter to his/her profile where he can see the reports from the users and taxi drivers, and make further actions, banning or not a user or taxi driver.



Img. 12 Admin Home Page

After the admin logs in the app, he/she has access to all the reports that a user or a driver has given. By selecting a particular report (both from the user or the driver) admin will be lead to a new page with the details of the report.



After clicking on a particular report (example report 1) the admin gain access to its details, such as user's profile or driver's profile, name, surname and email. Also he can see a description of the problem that happened. Therefore, he decides whether the report makes sense and bans the user/driver, or decline it.

3.2.2 API Interfaces

For the purpose of myTaxiService application Google Maps API (https://developers.google.com/maps) is used to represent the map around user’s current location, as well as to provide routes between two given addresses. Google Places API (https://developers.google.com/places) is used to provide the user with autocompleting addresses and to suggest addresses.

3.2.3 Hardware interfaces

There are no hardware interfaces that are used with this application.

3.2.4 Software interfaces

* Database Management System (DBMS): MySQL
* Programming technologies: HTML5, CSS3, jQuery, PHP
* Application server: Apache
* Operating System (OS): Cross platform

***4. Scenarios***

**Scenario 1:** Mike is a layer and he travels a lot, because his company has two main offices, one in Milan, and other in Rome. He has a flight on Wednesday night from Milan to Rome, and his colleague tells him about a new application called myTaxiService, that he has already used and was very satisfied from it. Mike decides to use the service of the app and he makes a profile. After that, he makes reservation for Wednesday night in order to go to the airport. His flight is at 10 pm so he makes reservation for 8 pm the same day, giving all the necessary information the app is requiring.

On Wednesday at 7:50 pm the taxi is in front of Mike's home and drives him to the destination.

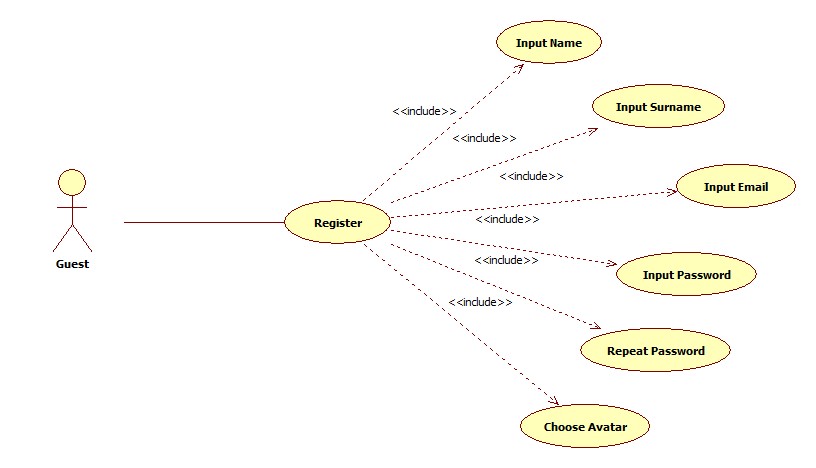
**Scenario 2**: Lina is preparing for a birthday party, but she realized that she is late. She remembers that she has installed on her smart phone an app that she saw on Facebook. The app name was myTaxiService. She already has created a profile but this is her first time using the service of the app. She requests a taxi given her current location, and ending destination. After a minute she receives a notification that the taxi with 005 code will be there in 10 minutes.

Lina is in front of her building waiting for the car, and already has past 20 minutes. The taxi driver is late. After 5 minute the taxi car arrives, and drove Lina to the birthday party. Lina is not satisfied with the service and she decides to report the taxi driver for the delay. Using the code of the taxi she sends report to the system.

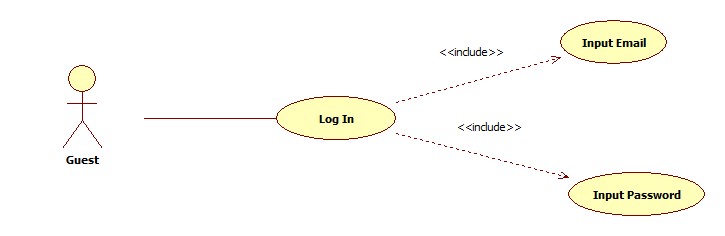
**Scenario 3:** Tom is a taxi driver and he uses myTaxiService app for better performance of his job. Right now, Tom has received 2 requests for a taxi in the zone that he is working, and one reservation 3 hours from now. He accepts the reservation because he is free at that time, and he also accepts just one of the requests, while he rejects the other. After accepting, the system sends notifications to the users.

5. UML Models

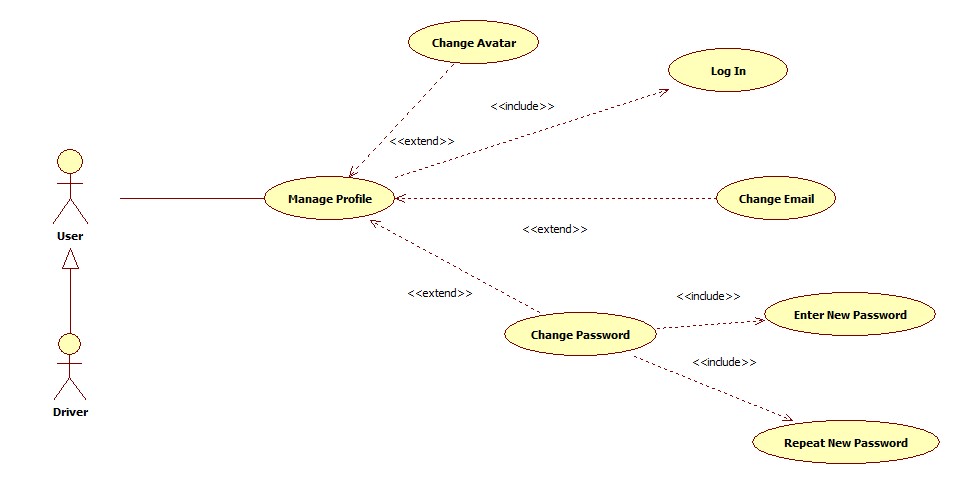
Use Case Diagrams



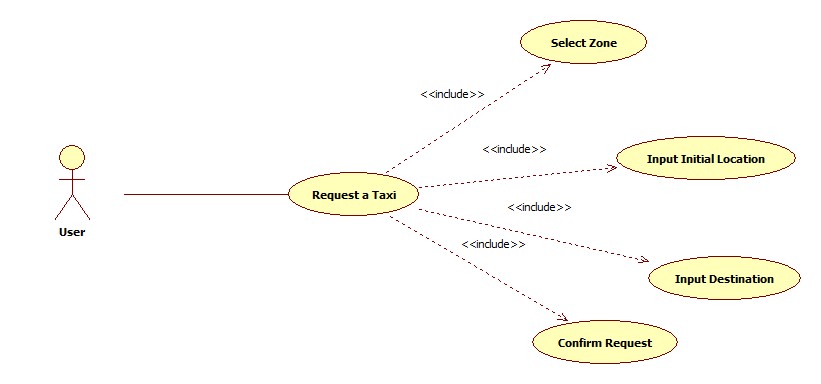
1. Register



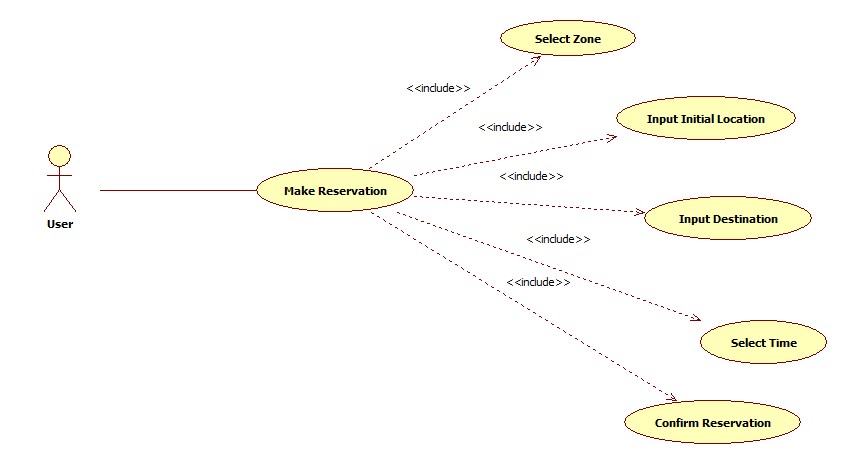
2. Log In



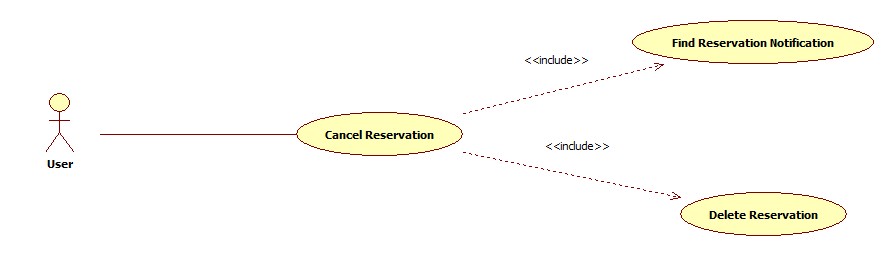
3. Manage Profile



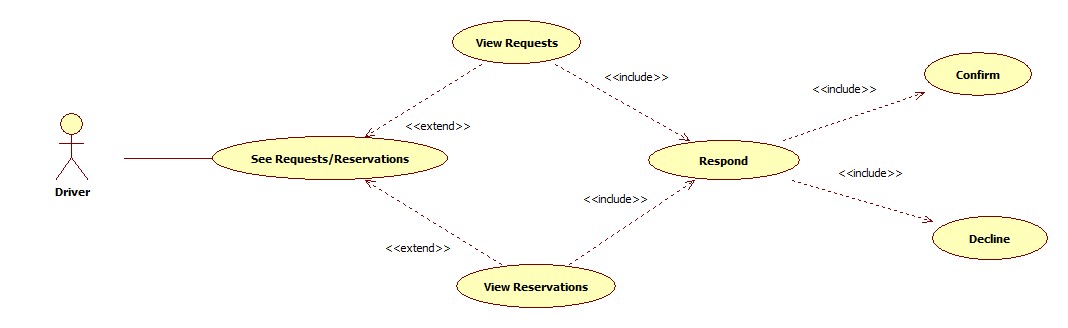
4. Request a Taxi



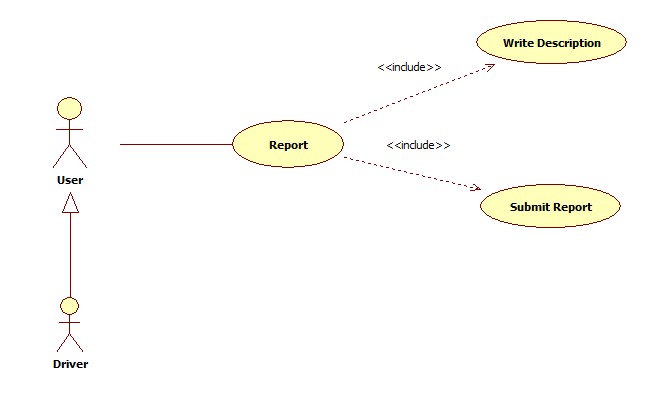
5. Make Reservation



6. Cancel Reservation



7. Driver`s Respond to a Request or Reservation



8. Report a User/Driver